

IN THE CLAIMS

This listing of the claim will replace all prior versions and listings of claim in the present application.

Listing of Claims

Claims 1-32 (canceled).

33. (new): A remote procedure call (RPC) optimizing method for optimizing RPCs between a server offering at least one remote procedure, and a client carrying out processing by use of a RPC calling said at least one remote procedure, on a computer for executing at least one of a program or a program part, when an interface definition language (IDL) description for said at least one remote procedure is provided, said remote procedure call optimizing method comprising the steps of:

analyzing a source code of said client so as to detect a remote procedure execution sequence constituting a set of RPCs that are highly likely to be executed successively;

determining a new remote procedure for executing said remote procedure execution sequence in a single RPC; and

defining an interface of said new remote procedure having been determined into said IDL description so as to allow said client to call said new remote procedure.

34. (new): A remote procedure call optimizing method for optimizing RPCs between a server offering at least one remote procedure, and a client carrying out processing by use of a RPC calling said at least one remote procedure, on a

computer for executing at least one of a program or one program part, when an IDL description for said at least one remote procedure is provided, said remote procedure call optimizing method comprising the steps of:

inputting a source code of said client and said IDL description;

detecting from said client source code a processing part including at least two RPCs issued in close proximity to one another;

calculating a first procedure causing said server to carryout a process equivalent to said processing part;

replacing said processing part of said client source code with a call to said first procedure, thereby generating a new client source code;

adding an interface of said first procedure to said IDL description, thereby generating a new IDL source code; and

adding a definition of said first procedure to a source code of said server.

35. (new): A remote procedure call optimizing method for optimizing RPCs between a server offering at least one remote procedure, and a client carryout out processing by use of a RPC calling said at least one remote procedure, on a computer for executing at least one of a program or a program part, when an IDL description for said at least one remote procedure is provided, said remote procedure call optimizing method comprising the steps of:

inputting a source code of said client and said IDL description;

detecting from said client source code a processing part including at least two RPCs issued in close proximity to one another;

calculating a first procedure causing said server to carryout a process equivalent to said processing part;

replacing said processing part of said client source code with a call to said first procedure, thereby generating a new client source code;

adding an interface of said first procedure to said IDL description, thereby generating a new IDL source code; and

generating an additional server source code for accommodating connection between said interface of said first procedure and an original source code of said server.

36. (new) A remote procedure call optimizing method according to claim 35, further comprising the steps of:

inputting said new IDL source code to an IDL compiler which in turn outputs a client stub, an RPC header file and a server stub;

compiling and linking said client stub, said RPC header file and said server stub, as well as said new client source code of each client and said additional server source code of each server, thereby outputting a client object and an additional server object;

dynamically linking said additional server object and a server object to output a new server object;

executing said client object on a first computer;

transmitting an RPC from said client object in order to execute said client object on said first computer; and

executing on a second computer said new server object having received said RPC.

37. (new): A remote procedure call optimizing method according to claim 34, further comprising the steps of:

inputting said new IDL source code to an IDL compiler which in turn outputs a client stub, an RPC header file and a server stub; and

compiling and linking said client stub, said RPC header file and said server stub, as well as said new client source code of each client and said new server source code of each server, thereby outputting a client object and a server object.

38. (new): A remote procedure call optimizing method according to claim 35, further comprising the steps of:

compiling said new client source code and said additional server source code so as to output a client class code and an additional server class code which can each be executed by a Java virtual machine (JVM);

compiling the additional server class code using a remote method invocation (RMI) compiler in Java so as to output a client stub and a server stub;

processing the client class code and the client stub using a first JVM; and

executing the additional server class code and the server stub using a second JVM.

39. (new): A remote procedure call optimizing method according to claim 35, further comprising the steps of:

compiling said new client source code and said new server source code so as to output a client class code and a server class code which can each be executed by a Java virtual machine (JVM);

compiling an output additional server class code using a remote method invocation (RMI) compiler in Java so as to output a client stub and a server stub;

processing the client class code and the client stub using a first JVM; and

executing the additional server class code and the server stub using a second JVM.
